

Referencia Rápida de XPath

Ejes, predicados, funciones, operadores, selección de nodos

Sintaxis

Expresiones de Ruta

/	Nodo raíz (inicio de ruta absoluta)
/bookstore/book	Selección de hijo directo
//book	Seleccionar todos los nodos book en cualquier parte
.	Nodo de contexto actual
..	Padre del nodo actual
@lang	Atributo llamado lang
node()	Cualquier nodo de cualquier tipo
*	Cualquier nodo elemento
@*	Cualquier atributo

Ejemplos Básicos

```
/html/body/div # ruta absoluta a <div>

```

Combinar Rutas

```
//book/title | //book/price # unión de dos rutas
//h1 | //h2 | //h3 # múltiples tipos de elemento
```

Ejes

Direcciones de Eje

child::	Hijos directos (eje predeterminado)
parent::	Padre directo
ancestor::	Todos los ancestros hasta la raíz
ancestor-or-self::	Ancestros + nodo actual
descendant::	Todos los descendientes
descendant-or-self::	Descendientes + nodo actual
following::	Todos los nodos después del actual en el documento
following-sibling::	Hermanos después del actual
preceding::	Todos los nodos antes del actual en el documento
preceding-sibling::	Hermanos antes del actual
self::	Solo el nodo actual
attribute::	Atributos del nodo actual
namespace::	Nodos de espacio de nombres

Ejemplos de Eje

```
//div/child::p # hijos <p> de <div>
//td/parent::tr # padre <tr> de <td>
//h2/following-sibling::p # <p> después de <h2>
//li/ancestor::ul # <ul> que contiene <li>
```

Predicados

Filtrar con Predicados

```
//book[1] # primer elemento book
//book[last()] # último elemento book
//book[position() < 3] # primeros dos books
//book[@lang='en'] # books con lang="en"
//book[price > 30] # books con precio > 30
```

Patrones de Predicado

[n]	Elemento en posición n (base 1)
[last()]	Último elemento
[last()-1]	Penúltimo
[@attr]	Tiene atributo
[@attr='val']	Atributo igual a valor
[element]	Tiene elemento hijo
[element='text']	Elemento hijo contiene texto
[not(@attr)]	No tiene atributo

Predicados Encadenados

```
//div[@class='list']//a[1] # primer <a> en div.list

```

Funciones

Funciones de Cadena

contains(s, sub)	Verdadero si s contiene sub
starts-with(s, pre)	Verdadero si s comienza con pre
string-length(s)	Longitud de la cadena
normalize-space(s)	Recortar y colapsar espacios en blanco
concat(a, b, ...)	Unir cadenas
substring(s, pos, len)	Extraer subcadena (base 1)
translate(s, from, to)	Reemplazo carácter por carácter

Funciones Numéricas

sum(node-set)	Suma de valores numéricos
count(node-set)	Número de nodos
floor(n)	Redondear hacia abajo
ceiling(n)	Redondear hacia arriba
round(n)	Redondear al entero más cercano
number(val)	Convertir a número

Ejemplos de Funciones

```
//div[contains(@class, 'active')]
//a[starts-with(@href, 'https')]
//p[string-length(text()) > 100]
//ul[count(li) > 5]
```

Operadores

Operadores de Comparación

=	Igual
!=	No igual
<	Menor que
<=	Menor o igual que
>	Mayor que
>=	Mayor o igual que

Lógicos y Aritméticos

and	AND lógico
or	OR lógico
not()	NOT lógico (función)
+	Suma
-	Resta
*	Multiplicación
div	División
mod	Módulo
	Unión de conjuntos de nodos

Ejemplos de Operadores

```
//book[price > 20 and price < 50]
//item[@type='a' or @type='b']
//span[not(contains(@class, 'hidden'))]
```

Pruebas de Nodo

Tipos de Nodo

node()	Cualquier nodo (elemento, texto, comentario, PI)
text()	Solo nodo de texto
comment()	Solo nodo de comentario
processing-instruction()	Nodo de instrucción de procesamiento
*	Cualquier nodo elemento
@*	Cualquier nodo atributo
element-name	Elemento con nombre específico

Ejemplos de Prueba de Nodo

```
//p/text() # contenido de texto de <p>
//div/comment() # comentarios dentro de <div>
//body/node() # todos los nodos hijo de <body>
//div/* # todos los hijos elemento de <div>
```

Funciones Booleanas

true()	Booleano verdadero
false()	Booleano falso
boolean(expr)	Convertir a booleano
not(expr)	Negar booleano
lang(code)	Verdadero si el idioma del nodo coincide

Abreviaciones

Forma Corta vs Larga

(ninguna)	child:: (eje predeterminado)
@	attribute::
//	/descendant-or-self::node()/
.	self::node()
..	parent::node()
[n]	[position()=n]

Ejemplos Abreviados

```
# Estos pares son equivalentes:
child::div -> div
attribute::href -> @href
/descendant-or-self::node()/p -> //p
self::node() -> .
parent::node() -> ..
```

Patrones Abreviados Comunes

```
//div[@id='main'] # div con id="main"
//table//td # todos los <td> en cualquier <table>
../sibling # hermano vía padre
//span # descendientes span del contexto
```

Patrones Comunes

Web Scraping / Testing

```
//input[@name='username'] # input de formulario por nombre
//button[text()='Submit'] # botón por texto
//div[contains(@class, 'error')] # elemento por clase parcial
//a[contains(@href, 'login')] # enlace por href parcial
```

Referencia Rápida de XPath

Selección Condicional

```
//div[@class='item']/span[@class='price']  
//tr[td[1]='Active'] # fila donde 1ra celda = Active  
//*[contains(text(), 'Warning')] # cualquier elemento con texto
```

XPath en Python (lxml)

```
from lxml import html  
tree = html.fromstring(page_content)  
links = tree.xpath('//a/@href')  
titles = tree.xpath('//h2/text()')
```

XPath en Selenium

```
driver.find_element(By.XPATH, "//input[@id='search']")  
driver.find_elements(By.XPATH, "//li[@class='result']")
```